The Arizona Connected Vehicle Test Bed - SMARTDrive™, Anthem AZ

Larry Head
Systems and Industrial Engineering, College of Engineering
Arizona Connected Vehicle Research Program

• Test Bed for Initial Concept and Develop Research
• 2007 Initiated Connected Vehicle Research Program
  – Emergency First Responder

• Multi-Modal Intelligent Traffic Signal System
• Pooled Fund Project
  – FHWA, Maricopa County DOT, CALTRANS, VADOT, MnDOT, TxDOT, FDOT

• Traffic Signal Control in a Connected Vehicle Environment (MMITSS)
  – Intelligent Signal Control
  – Emergency Vehicle Priority/Preemption
  – Transit Priority
  – Freight Priority
  – Pedestrian Assistance

• Vehicle-to-Infrastructure Applications
• Connected Vehicle Work Zone: Innovative Technology Deployment (ITD) Grant Program
Arizona Connected Vehicle Test Bed (Maricopa County)

**Anthem, AZ**

DSRC Installations:
11 Signalized Intersection
6 Freeway Interchanges
10 Freeway Locations

Approximately 25,000 Residents
10,000 Vehicles
Arizona Connected Vehicle Test Bed
Application Testing

1. Functional Testing
2. Performance Assessment

Scenarios
- What do you want to test?
- How do we know it is working?

Simulation Testing
- Hardware/Software in the Loop

Field Testing
Testing Approach

Given a feature, concept, algorithm, etc.....

1. Develop the system doing simple software tests
2. Simulate operational scenarios to test **Functionality**
3. Execute simulation experiments to determine **Performance**
4. Perform field testing in Tucson (dummy intersection, signal controller, live in street)
5. Perform field testing in Anthem (acceptance)
6. Integrate into demonstrations (regression testing)
Connected Vehicle Applications

- Multi-Modal Intelligent Traffic Signal System (MMITSS)
  - Priority for EV, Transit, Trucks (EVP, TSP, FSP)
  - Intelligent Signal Control (I-SIG)
  - Pedestrian Application (PED-SIG)
- V2I Applications
  - Roadside Alert (RSA) – Work Zone, Incident, School Zone
- V2P Applications
  - Pedestrian in Crosswalk Warning (Conflict)
- V2V Applications
  - Emergency Vehicle Alert (EVA)
Arizona Connected Vehicle Test Bed
Application Testing

Field Testing Philosophy:
- Travel time to the test bed > 2 Hours
  - Be sure we are ready to test
- Field testing is expensive:
  - Signal Technician(s)
  - Vehicles (2-3++)
  - Time (real-time)
- Tests should be boring (known outcome)
Scenario Based Test Plans

- Test Emergency Vehicle Alert
  1. Stage Vehicles to arrive at the desired time
  2. Enable EVA message broadcast on each EV
  3. Validate that the message is received by other vehicles (EV, non-EV)
     i. Timeliness (when should it be receive?)
     ii. Multiple vehicle messages

- Test TSP and EVP at the same time
  1. Stage Vehicles to arrive at the desired time
  2. Priority requests from each vehicle
  3. Does each vehicle receive the MAP message?
  4. Does each vehicle compute their position on the MAP correctly?
  5. Does each vehicle send a SRM
  6. Does each vehicle receive a SSM (with the other vehicle data)?
  7. Is the “correct” signal timing determined by MMITSS?
     i. Base on the initial status of the signal, timing, and arrival times
Simulation Based Testing

SIL/CID

VISSIM

driverModel.dll

VISSIM x-y plane to GPS (WGS - 84)
Vehicle Position Mapping

OBE #m

OBE

OBE

OBE

OBE Distributor

RSE

RSE #n

RSE

MMITSS

Signal Controller

Ethernet - NTCIP 1202

In-Vehicle Display (HIL)

Ethernet or DSRC (5.9 GHz) [SAE J2735]

GID #n
• Create connected vehicle types and compositions in VISSIM

<table>
<thead>
<tr>
<th>No</th>
<th>Name</th>
<th>Category</th>
<th>Model2D3DDist</th>
<th>ColorDist1</th>
<th>OccupDist1</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Car</td>
<td>Car</td>
<td>10: Car</td>
<td>85: Blue</td>
<td>1: Occupancy 1.00</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>HGV</td>
<td>HGV</td>
<td>20: HGV</td>
<td>89: Black</td>
<td>1: Occupancy 1.00</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Bus</td>
<td>Bus</td>
<td>30: Bus</td>
<td>87: Purple</td>
<td>1: Occupancy 1.00</td>
<td>45</td>
</tr>
<tr>
<td>4</td>
<td>Tram</td>
<td>Tram</td>
<td>40: Tram</td>
<td>1: Default</td>
<td>1: Occupancy 1.00</td>
<td>9999</td>
</tr>
<tr>
<td>5</td>
<td>Pedestrian</td>
<td>Pedestrian</td>
<td>50: Pedestrian</td>
<td>1: Default</td>
<td>1: Occupancy 1.00</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>Bike</td>
<td>Bike</td>
<td>60: Bike</td>
<td>1: Default</td>
<td>1: Occupancy 1.00</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>ConnectedVeh</td>
<td>Car</td>
<td>10: Car</td>
<td>85: Red</td>
<td>1: Occupancy 1.00</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>ConnectedHGV</td>
<td>HGV</td>
<td>20: HGV</td>
<td>90: White</td>
<td>1: Occupancy 1.00</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>ConnectedBus</td>
<td>Bus</td>
<td>30: Bus</td>
<td>88: Yellow</td>
<td>1: Occupancy 1.00</td>
<td>45</td>
</tr>
</tbody>
</table>

• Use drivermodel.dll for connected vehicle types
Simulation Platform

- **DriverModel.dll**

```c

// Get vehicle local coordinates from VISSIM

switch (DRIVER_DATA_VEH_REAR_X_COORDINATE) {
    case DRIVER_DATA_VEH_REAR_X_COORDINATE:
        veh_rear_x = double_value;
        return 1;
    case DRIVER_DATA_VEH_REAR_Y_COORDINATE:
        veh_rear_y = double_value;
        dx = vehicle_x - veh_rear_x;
        dy = vehicle_y - veh_rear_y;
        veh_heading = atan2(dy, dx) * 180.0 / PI;
        x_cal = (vehicle_x + veh_rear_x) / 2;
        y_cal = (vehicle_y + veh_rear_y) / 2;
        break;
}

g_ePoint.local2ecef(y_cal, x_cal, 0.0, &x_grid, &y_grid, &z_grid); // Third argument z=0.0 for viess

g_ePoint.cccf2lls(x_grid, y_grid, z_grid, &g_long, &g_lat, &g_altitude);

if (veh_heading < 0)
    veh_heading = veh_heading + 360.0;

dsecond = (long)(vehicle_time_new * 10) % 100; // TO GET THE milli second of current simulation time

CreateVehMsg(vehicle_id_new, dsecond, msgCnt, g_lat, g_long, veh_heading, veh_speed);

veh.Vehicle2BSM(blobOut);

if ((temp_int = sendto(socketsc_connect, blobOut, BSM_MSG_SIZE, 0, (SOCKADDR*)&a, sizeof(SOCKADDR_IN)) != BSM_MSG_SIZE))
    { 
        fprintf(fp,"send error... %d\n", temp_int);
    }

```

- **Transform (center of the vehicle) to GPS coordinates**

- **Pack to BSM**

- **Send BSM through UDP**
OBE Distributor

• Required to route BSMs to the appropriate RSE
  – Matches vehicle position to (approaching) RSEs within DSRC range
  – The DSRC range of each RSE is configurable
  – RSE’s have unique IP addresses (e.g. 10.1.1.101)

• Model GPS Errors

• Additional OBE Logic for priority eligible vehicles (Priority Request Generator)
  – Signal Request Message (SRM) management
    • Generate new SRM, update SRM, cancel SRM,...
VISSIM (ver 6) Simulation Network
Calibration:
DSRC Range of Intersections and MAP Data
Software-in-the-loop Simulation (Docker)

Windows PC – VISSIM
IP: 10.254.56.2

Virtual Controller 1
IP: 10.254.56.2
Server IP: 10.254.56.201
PORT: 501

Virtual Controller 2
IP: 10.254.56.2
Server IP: 10.254.56.202
PORT: 502

Virtual Controller 10
IP: 10.254.56.2
Server IP: 10.254.56.210
PORT: 510

Linux PC
IP: 10.254.56.8

Docker Container 1
IP: 10.254.56.201

Docker Container 2
IP: 10.254.56.202

Docker Container 10
IP: 10.254.56.210

Application 1
Application 2
.
.
.
Application n

Application 1
Application 2
.
.
.
Application n

Application 1
Application 2
.
.
.
Application n

Application 1
Application 2
.
.
.
Application n

Config File 1
Config File 2
.
.
.
Config File n

Config File 1
Config File 2
.
.
.
Config File n

Config File 1
Config File 2
.
.
.
Config File n

Config File 1
Config File 2
.
.
.
Config File n

Config File 1
Config File 2
.
.
.
Config File n

Config File 1
Config File 2
.
.
.
Config File n

...
...
...

IP: 10.254.56.2
Server IP: 10.254.56.202
PORT: 502
IP: 10.254.56.2
Server IP: 10.254.56.201
PORT: 501
IP: 10.254.56.2
Server IP: 10.254.56.210
PORT: 510
...
...
...
Field Testing Scenarios, March 3rd and 4th, 2016: Designed and Conducted by Leidos (IA Contractor)

- 2 trucks with priority in northbound/southbound
- 2 buses with priority in eastbound/westbound
- 10 rounds of testing
- 6 regular vehicles

Source: Leidos Field Test Plan
Field Test Result for Transit Priority

Transit #1

<table>
<thead>
<tr>
<th>Round</th>
<th>TT (sec) With Priority</th>
<th>TT (sec) Without Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>420</td>
<td>490</td>
</tr>
<tr>
<td>2</td>
<td>480</td>
<td>550</td>
</tr>
<tr>
<td>3</td>
<td>540</td>
<td>610</td>
</tr>
<tr>
<td>4</td>
<td>600</td>
<td>670</td>
</tr>
<tr>
<td>5</td>
<td>660</td>
<td>730</td>
</tr>
<tr>
<td>6</td>
<td>720</td>
<td>790</td>
</tr>
<tr>
<td>7</td>
<td>780</td>
<td>850</td>
</tr>
<tr>
<td>8</td>
<td>840</td>
<td>910</td>
</tr>
<tr>
<td>9</td>
<td>900</td>
<td>970</td>
</tr>
<tr>
<td>10</td>
<td>960</td>
<td>1030</td>
</tr>
</tbody>
</table>

 Average TT (sec) = 850.12
 TT Standard Deviation = 91.13

Transit #2

<table>
<thead>
<tr>
<th>Round</th>
<th>TT (sec) With Priority</th>
<th>TT (sec) Without Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>480</td>
<td>550</td>
</tr>
<tr>
<td>2</td>
<td>540</td>
<td>610</td>
</tr>
<tr>
<td>3</td>
<td>600</td>
<td>670</td>
</tr>
<tr>
<td>4</td>
<td>660</td>
<td>730</td>
</tr>
<tr>
<td>5</td>
<td>720</td>
<td>790</td>
</tr>
<tr>
<td>6</td>
<td>780</td>
<td>850</td>
</tr>
<tr>
<td>7</td>
<td>840</td>
<td>910</td>
</tr>
<tr>
<td>8</td>
<td>900</td>
<td>970</td>
</tr>
<tr>
<td>9</td>
<td>960</td>
<td>1030</td>
</tr>
<tr>
<td>10</td>
<td>1020</td>
<td>1090</td>
</tr>
</tbody>
</table>

 Baseline (2 buses without Priority Requests for 10 Round Trips)
 Average TT (sec) = 850.12
 TT Standard Deviation = 91.13

 TSP (2 buses with Priority Requests for 10 Round Trips)
 Average TT (sec) = 762.56
 TT Standard Deviation = 53.48

 Improvement (%)

<table>
<thead>
<tr>
<th>Improvement (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10.3</td>
</tr>
<tr>
<td>-41.3</td>
</tr>
</tbody>
</table>
Tools to Support Testing

- Run Time Data
  - Logs (files from the applications)
  - Time-Phase Diagrams
Field Testing starts with good lab (simulation and bench) testing
- Scenarios define the test parameters
- Field testing is hard, slow, and hopefully, boring
- Performance assessment using simulation is necessary to estimate impacts
Larry Head
Professor of Systems and Industrial Engineering
Director of the UA Transportation Research Institute
klhead@email.arizona.edu